

# **Platforma Microsoft Surface**

## **Microsoft Surface Platform**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2010

.....

Rád bych na tomto místě poděkoval vedoucímu mé bakalářské práce panu Ing. Michalovi Radeckému, za odborné vedení a cenné rady.

## Abstrakt

Bakalářská práce se zabývá novou platformou Microsoft Surface. Práce je teoreticko-aplikačního charakteru. V teoretické části je uveden nejdříve celkový rozbor této platformy, na který navazuje úvod do vývoje aplikací pro tuto platformu.

V porovnání s programy pro běžné počítače jsou zde nemalé rozdíly nejen ve vzhledu, ale i ovládání, a proto jsou v práci uvedena specifika vývoje pro tuto platformu. Práce se také zamýšlí nad možnou budoucností této technologie.

Aplikační část se skládá z implementace ukázkové aplikace a její analýzy. Jedná se o jednu aplikaci implementující většinu profilových vlastností této platformy.

Díky mnoha možnostem, které nabízí a podpory jedné z největších softwarových společností, se stává vývoj takových aplikací perspektivním oborem. A tato bakalářská práce přispívá k rychlému zorientování se a snadnějšímu začátku práce s touto novou technologií.

**Klíčová slova:** Microsoft Surface, multitouch, tag, Člověče nezlob se

## Abstract

Bachelor's work deals with a new platform Microsoft Surface. The work is of both theoretical and practical character. In theoretical section, there is a complete analysis of the platform at first, at which builds introduction to development of applications for this platform.

Compared to programmes for casual computers, there are considerable differences in appearance and control, therefore specifics of development for this platform are included in the work. The work thinks about the possible future of this technology.

Practical section is consisted of sample application and its analysis. It is a one application implementing most of the main features of this platform.

Thanks to the many possibilities and support of one of the biggest software companies, the development of such applications is becoming a promising field. And this bachelor's work, helps to quick and better beginning of working with this new technology.

**Keywords:** Microsoft Surface, multitouch, tag, Mensch ärgere dich nicht

## Seznam použitých zkratk a symbolů

2D	– Two Dimensional
3D	– Three Dimensional
API	– Application Programming Interface
AD&D	– Advanced Dungeons and Dragons
AT&T	– American Telephone and Telegraph
CPU	– Central Processing Unit
DLP	– Digital Light Processing
GUI	– Graphical User Interface
HDTV	– High-Definition Television
HWND	– Handle Window
LED	– Light-Emitting Diode
MS	– Microsoft
MSDN	– Microsoft Developer Network
MSNBC	– Microsoft National Broadcasting Company
OS	– Operating System
PC	– Personal Computer
RAM	– Random-Access Memory
RFID	– Radio Frequency Identification
SDK	– Software Development Kit
SP	– Service Pack
UI	– User Interface
WiFi	– Wireless Fidelity
WPF	– Windows Presentation Foundation
XNA	– XNA's Not Acronymed
XAML	– Extensible Application Markup Language

## Obsah

<b>1</b>	<b>Úvod</b>	<b>4</b>
<b>2</b>	<b>Koncept Microsoft Surface</b>	<b>5</b>
2.1	Co je to Microsoft Surface . . . . .	5
2.2	Možnosti Microsoft Surface . . . . .	5
2.3	Funkcionalita . . . . .	6
2.4	Architektura . . . . .	8
<b>3</b>	<b>Programování pro Microsoft Surface</b>	<b>11</b>
3.1	Prerekvizity . . . . .	11
3.2	Surface simulátor . . . . .	12
3.3	Vstupy . . . . .	13
3.4	Vývoj aplikací ve vrstvě jádra . . . . .	16
3.5	Vývoj aplikací v prezentační vrstvě . . . . .	16
<b>4</b>	<b>Implementace ukázkové aplikace</b>	<b>18</b>
4.1	Analýza hry . . . . .	18
4.2	Popis aplikace . . . . .	18
4.3	Analýza aplikace . . . . .	20
4.4	Využití SDK . . . . .	21
4.5	Možnosti vylepšení aplikace . . . . .	21
<b>5</b>	<b>Specifika vývoje pro Microsoft Surface</b>	<b>25</b>
5.1	Tvorba GUI . . . . .	25
5.2	Srovnání prezentační vrstvy a vrstvy jádra . . . . .	29
5.3	Ošetření vstupů . . . . .	29
<b>6</b>	<b>Budoucnost Microsoft Surface</b>	<b>30</b>
<b>7</b>	<b>Závěr</b>	<b>31</b>
<b>8</b>	<b>Reference</b>	<b>32</b>
	<b>Přílohy</b>	<b>32</b>
<b>A</b>	<b>Class Diagram</b>	<b>33</b>

## Seznam obrázků

1	Microsoft Surface[2]	5
2	Využití Microsoft Surface	6
3	Snímek povrchu[4]	7
4	Průřez Microsoft Surface	8
5	Architektura Microsoft Surface	9
6	Povinné požadavky	12
7	Microsoft Surface simulátor	13
8	Blokové schéma vstupů	15
9	Ukázka tagů	16
10	Aplikace Člověče nezlob se!	18
11	Diagram hry	19
12	Úvodní obrazovka	20
13	Diagram akcí hodu kostky	23
14	Diagram akcí posunu figurky	24
15	Posunutí fotografie	26
16	Zvětšení fotografie	27
17	Class Diagram	33

## Seznam tabulek

1	Prostředí simulátoru . . . . .	14
2	Ovládací prvky prezentační vrstvy . . . . .	17
3	Použité tagy . . . . .	20



## 1 Úvod

Moderní technologie se dostaly do bodu, kdy klasická uživatelská rozhraní už nestačí. I zde se uplatňuje pravidlo: „Čím jednodušší, tím lepší“. A co je pro člověka přirozenější než neverbální komunikace, jejíž součástí je gestikulace rukou.

Jednou takovou technologií je i MS Surface, který se ovládá právě pomocí přímého doteku rukou a jejich gest, či interakcí s jinými objekty. Běžnému uživateli tak odpadá starost se správou vstupních zařízení a učením se používat je.

Cílem této práce je poskytnout jakýsi manuál hlavně pro vývojáře začínající s touto platformou. Jako všichni začátečníci se nejdříve setkají s platformou z pohledu běžného uživatele, dozví se o tom jaké možnosti tato platforma nabízí, a jak vypadá v provozu.

Dále se seznámí s její architekturou a měli by poznat, jak tato platforma a její části fungují. Na řadu pak přichází fáze přípravy pro vývoj, tzn. co všechno budou potřebovat, aby mohli úspěšně začít. Seznámí se také se způsobem testování aplikací, i s tím že není nutné vlastnit přímo Microsoft Surface.

Nadále jsou definovány všechny způsoby, kterými se dají aplikace této platformy ovládat. Následuje rozbor dvou základních vrstev, určujících druh aplikace.

Pro rychlejší uvedení do tématu je zahrnuta ukázková aplikace spolu s její analýzou a zdrojovými kódy. Aby se vývojáři vyvarovali některých zvyků z programování desktopových aplikací, je zde uveden výčet základních rad pro návrh aplikací a také odlišnosti této platformy, které musí být brány v potaz.

Závěrem je shrnuta pravděpodobná budoucnost této platformy, jinými slovy proč se vlastně učit vyvíjet aplikace pro tuto platformu.

## 2 Koncept Microsoft Surface

### 2.1 Co je to Microsoft Surface

Microsoft Surface je hardwarová a softwarová platforma pro vývoj dotekových aplikací, umožňující zpracování více vstupů najednou. Platforma umožňuje vytvářet bohaté a vizuálně působivé aplikace, nabízející nové uživatelské zkušenosti, s ovládáním jen pomocí rukou nebo různých objektů.

Microsoft Surface aplikace by měla být přirozená a intuitivní. Neměla by vykazovat skoro žádnou podobnost s tradičními Microsoft Windows nebo Web aplikacemi. Uživatel by ve skutečnosti neměl vědět, že je v jednotce Microsoft Surface počítač.[1]

To totiž nemusí být na první pohled zřejmé, protože Microsoft Surface byl navržen v podobě konferenčního stolku (obr. 1). Jeho rozměry jsou 53 x 107 cm, s výškou 56 cm a váhou kolem 100 kg. Vrchní deska je vyrobena z akrylátu. Větší část této desky se během provozu mění na multidotykový displej. Kostra stolku je vyrobena z práškem obalované oceli.



Obrázek 1: Microsoft Surface[2]

### 2.2 Možnosti Microsoft Surface

Microsoft Surface je platforma, která reaguje na přirozené pohyby rukou a na interakci s reálnými objekty. Je vybaven 360 stupňovým uživatelským rozhraním a 30-ti palcovým multidotykovým displejem.

Pro srovnání iPhone, který bývá často uváděn jako vzor dotekového ovládání, současně snímá dva dotekové body. To je pro práci s ručním zařízením jako je iPhone dostačující, ovšem pro Surface, jehož ovládací plocha je řádově větší a může s ním tak pracovat několik uživatelů najednou, by tento přístup nebyl vyhovující. Proto byl Surface optimalizován na rozpoznání až 52 současných doteků.

Surface ovšem nereaguje jen na dotek rukou, ale i na interakci s libovolným objektem, u kterého rozpozná i tvar styčné plochy a orientaci objektu, prstu atd.

Dalším typem dotekových bodů jsou tagy (matrixové kódy). Podporovány jsou dva formáty, jeden o velikosti 1 byte, druhý o velikosti 128 bytů. Nalepeny na předmětech mohou sloužit jako jednoznačné identifikátory.[3]



Obrázek 2: Využití Microsoft Surface

Právě díky těmto vlastnostem už Surface našel mnohé uplatnění v různých službách (obr. 2). Například partnerské společnosti firmy Microsoft už používají Surface ve svých hotelech, restauracích a obchodních řetězcích. Surface slouží třeba návštěvníkům restaurací k výběru jídla nebo hostům v hotelech k plánování výletů po okolí. Jedním řetězcem hotelů „zaměstnávajícím“ Surface jsou Starwood Hotels, které plánují umožnit svým zákazníkům položením karty na stůl zaplatit za hudbu, knihy a další hotelové služby. Oproti tomu v obchodech síť AT&T se Surface používá k interaktivním prezentacím služeb a produktů, např. zákazník si může vybrat dva mobilní telefony, které položí na Surface, ten následně zobrazí srovnání a analýzu těchto přístrojů. Dalším kdo využil potenciálu Surface je stanice MSNBC, která jej umístila ve svém volebním studiu, kde během prezidentských voleb Spojených Států Amerických v roce 2008 poskytoval informace a analýzy probíhajících voleb.

### 2.3 Funkcionalita

V podstatě MS Surface funguje tak, že na vrchní akrylátovou desku promítá projektor aktuální stav GUI. Uživatel pak pomocí rukou či jiných objektů vytvoří jeden nebo více kontaktů v návaznosti na toto rozhraní.

O detekci kontaktů na povrchu se stará infračervený systém, který vysílá směrem k desce infračervené paprsky. Ty se od desky odrazí zpět (resp. odrazí se jen v místech, kde se nachází objekty, které jim brání v cestě, tzn. v místě kontaktu), kde jsou zachy-

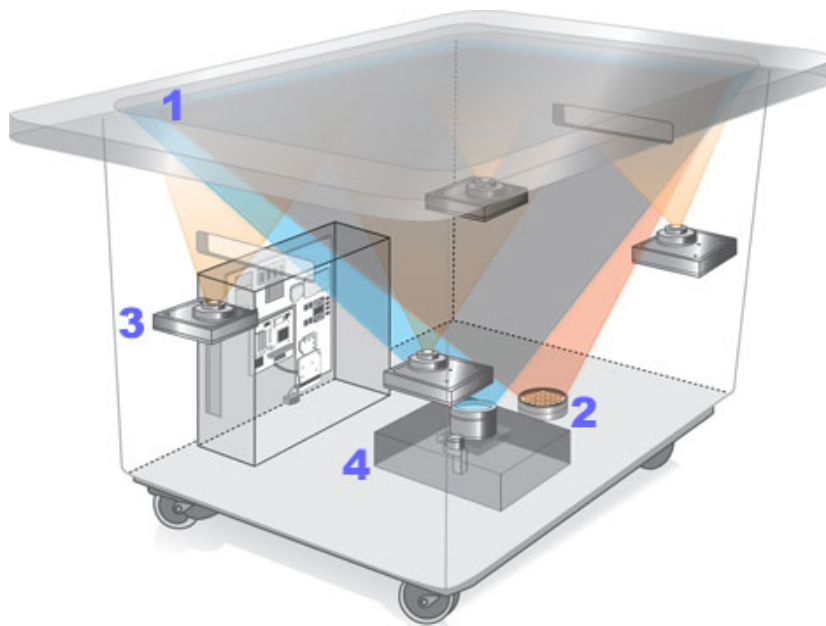
ceny snímači. Tímto způsobem se pořídí něco jako obrázek povrchu desky, připomínající rentgenový snímek (obr. 3). Tyto snímky (ve skutečnosti se zachytí několik snímků za sekundu, tudíž jde o video) zpracuje procesor a v závislosti na právě běžící aplikaci provede změny v GUI.



Obrázek 3: Snímek povrchu[4]

Na obrázku 4 je znároněn průřez jednotkou Microsoft Surface. Jednotlivé prvky platformy a jejich účel jsou detailněji popsány níže.

1. **Obrazovka** – difuzor proměňuje vrchní akrylátovou desku ve velkou horizontální „multidotekovou“ obrazovku, která je schopna zpracovat mnoho vstupů od více uživatelů. Surface dokáže rozeznat objekty podle jejich tvaru nebo přečtením kódovaných „domino“ tagů.
2. **Infračervený Systém** – tento systém pracuje v blízkém infračerveném spektru. Používá LED technologii s 850-ti nanometrovou vlnovou délkou. Ta vysílá paprsky směrem k vrchní desce. Když se objekty této desky dotknou, světlo se odrazí zpět a je detekováno několika infračervenými kamerami, které pokrývají síť o rozměrech 1280 x 960 bodů.
3. **Hardware** – Surface využívá mnoho komponent z běžných stolních počítačů — Core 2 Duo procesor, 2 GB RAM paměti a 256 MB grafickou kartu. Bezdrátová komunikace se zařízeními na povrchu je ošetřena pomocí WiFi a Bluetooth antén (budoucí verze mohou zahrnovat RFID nebo Near Field Communication). Základním operačním systémem je upravená verze Windows Vista.
4. **Projektor** – Microsoft Surface využívá ten samý světelný DLP engine jako většina HDTV projektorů. Stopa obrazovky viditelného světla na 1024 x 768 pixelů, je ve skutečnosti menší než překrývající se neviditelná infračervená projekce. To umožňuje lepší rozeznávání na hranách obrazovky.[5]



Obrázek 4: Průřez Microsoft Surface

## 2.4 Architektura

Jak vyplývá z obr. 5. platforma Microsoft Surface je komplexní hardwarová a softwarová technologie, která obsahuje několik specifických vlastností.

Základem této architektury je hardware, nad kterým operuje daný OS. Mezi OS a aplikacemi se nachází dva pilíře.

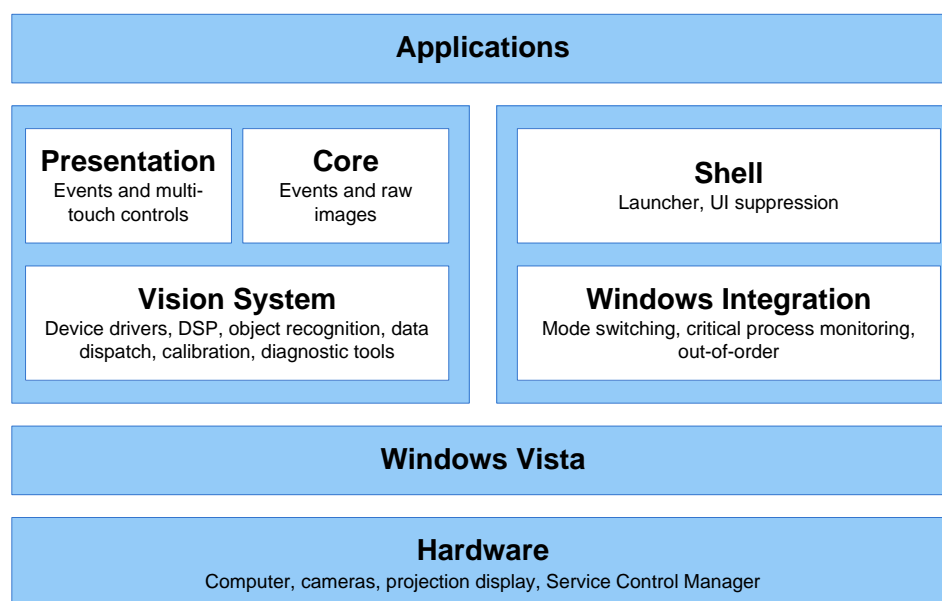
První pilíř se stará o celkové ovládání a fungování aplikací. Základ tvoří snímací systém, který vstupy od uživatele převede na data srozumitelná SDK. Ty jsou pak zpracovány prezentační vrstvou nebo vrstvou jádra. Záleží na tom v jaké vrstvě byla daná aplikace vytvořena. Tyto vrstvy pak na základě vstupních dat provedou naprogramované procedury, které se nakonec projeví v aplikaci.

Druhý pilíř má na starost správu systému. Jedná se hlavně o integraci Windows, a to tak aby zajistily bezchybný chod prostředí MS Surface a přepínání mezi uživatelským a administrátorským módem. Součástí pilíře je i shell, který se stará o spouštění prostředí MS Surface, ve kterém běží aplikace, a potlačení Windows.

### 2.4.1 Windows Vista

Microsoft Surface používá jako svůj základ operační systém Windows Vista. Ty mu poskytují veškerou administrativu, bezpečnost a adresářovou strukturu. Microsoft Surface pracuje ve dvou režimech.

- **Administrátorský mód** – určen pro administrátory a vývojáře. Umožňuje plný přístup k funkcím Windows.



Obrázek 5: Architektura Microsoft Surface

- **Uživatelský mód** – určen pro uživatele. Prostředí Windows je úplně potlačeno.

## 2.4.2 Snímací Systém

Video data zachycená hardwarem jsou zpracována softwarem snímacího systému. Ten je převede do datového formátu srozumitelného pro MS Surface SDK. Ty jsou pak přístupné skrze API tohoto SDK.

Tento systém je do značné míry závislý na osvětlení okolního prostředí (pod jiným světlem se jinak vyhodnotí styčné body), z toho důvodu byl vybaven nástrojem kalibrace. Ta se dělí na dva typy: základní a kompletní. Kompletní kalibrace se provádí při přesunu Microsoft Surface na nové místo. A základní kalibrace se provádí při velké změně světelných podmínek okolí.

## 2.4.3 Prezentační vrstva a vrstva jádra

Microsoft Surface SDK informuje aplikace o kontaktech překrývajících jejich okna. Jak uživatelé pokládají kontakty na displej a manipulují s nimi, Microsoft Surface SDK o tom uvědomuje aplikace a dává jim možnost k aktualizaci jejich uživatelského prostředí.

Aplikace mohou u každého kontaktu zjistit jeho umístění, natočení, hranici a hlavní okruž. Po kontaktech, které byly vytvořeny označenými objekty (tzn. ty, které mají ze spodu vtisknuty tagy), mohou aplikace požadovat jejich hodnotu.

Microsoft Surface SDK nabízí dva druhy API: prezentační vrstvu a vrstvu jádra. Při vývoji Microsoft Surface aplikace se může použít jen jedna z nich.

- **Prezentační vrstva** se propojuje s WPF. Obsahuje soubor předvytvořených ovládacích prvků pro Microsoft Surface.
- **Vrstva jádra** se dá použít ve spolupráci s jakýmkoliv frameworkem pro tvorbu uživatelských rozhraní.

#### **2.4.4 Shell**

Surface Shell je komponenta, která se stará o aplikace, okna a uživatelské relace, příp. o další funkce. Každá Microsoft Surface aplikace se musí integrovat do shellu MS Surface.

#### **2.4.5 Integrace Surface a Windows**

Spojení MS Surface a operačního systému MS Windows, umožňuje přístup k systémovým funkcím tohoto OS. Tyto funkce poskytují podporu unikátních vlastností MS Surface, jako např. spravování uživatelských relací, přepínání mezi uživatelským rozhraním Windows (administrátorský mód) a uživatelským prostředím (uživatelský mód), sledování kritických procesů MS Surface a ošetření závažných poruch.[1]

## 3 Programování pro Microsoft Surface

### 3.1 Prerekvizity

Stejně jako vývoj všech ostatních aplikací probíhá i tento na osobním počítači. V čem nastává rozdíl jsou nároky na vybavení tohoto PC (obr. 6). Ty začínají hned s operačním systémem a pokračují určením programovacího jazyku a jeho frameworků. Dále je zde nutnost instalace speciálního SDK. Toto jsou povinné požadavky, dále pak existují i jiné nástroje, urychlující vývoj. Seznam konkrétních požadavků je uveden níže.

- **32-bitová verze Windows Vista/Windows 7** – základem každého počítače je jeho operační systém, pokud chcete vyvíjet aplikace pro Microsoft Surface musíte mít nainstalovánu jednu z novějších verzí Windows (u Windows Vista je nutné mít nainstalován Service Pack 1). Programování ve Windows 7 je možné jen se Surface SDK SP1. V tuto chvíli jsou podporovány jen 32-bitové verze těchto OS, ale na internetu lze najít návod, jak nainstalovat Surface SDK i na 64-bitové verze. V případě že tak učiníte bude vám Surface simulátor s předvytvořenými aplikacemi bezchybně fungovat. Problém nastane až budete chtít zkompilovat a spustit vlastní aplikaci. V tom případě se po vytvoření prvního kontaktu ve vaší aplikaci vyvolá výjimka a aplikace skončí. Důvodem tohoto problému by mohla být kompilace 64-bitovým kompilátorem a následné spuštění v 32-bitové aplikaci. Potenciálním řešením by tak mohlo být vynucené spuštění Visual Studia v 32-bitovém režimu.
- **Microsoft Visual C# 2008 Express Edition/Microsoft Visual Studio 2008 (nebo vyšší)** – pro psaní jakékoliv aplikace si musíte zvolit programovací jazyk. Pro Microsoft Surface se dají psát programy jen v jazyce C#. Výše uvedené jsou oficiální prostředí implementující tento jazyk.
- **Microsoft XNA Framework Redistributable 2.0 (nebo vyšší)** – tento framework slouží k snadnému a rychlému vývoji aplikací s použitím 2D nebo 3D grafiky. Během instalace je implementován do prostředí Microsoft Visual Studia (nebo Express záleží co jste si vybrali), kde poté můžete začít s vývojem tohoto druhu aplikací.
- **Microsoft Surface SDK 1.0 (SP1), Workstation Edition** – jedná se o balíček, který po instalaci zpřístupní vývoj Microsoft Surface aplikací, a to z prostředí Microsoft Visual Studia (nebo Express), a nainstaluje na danou stanici také simulátor pro testování vytvářených programů. Momentálně lze pořídit i Service Pack 1, který přináší mnohá vylepšení (např. možnost vyvíjení aplikací ve Windows 7).
- **Microsoft Expression Blend 2 (nebo vyšší)** – toto prostředí není povinné pro vývoj aplikací určených pro platformu Microsoft Surface. Týká se jen aplikací využívajících grafického systému WPF (tzn. aplikací postavených na prezentační vrstvě). Jedná se o velice efektivní nástroj pro rychlé a snadné vytváření GUI.





Obrázek 6: Povinné požadavky

### 3.2 Surface simulátor

Testování aplikací je umožněno dvěma způsoby. Jedním z nich je testování přímo na jednotce Microsoft Surface. Druhým způsobem je testování na běžném PC.

Vzhledem k tomu, že je Microsoft Surface dražší než běžné počítače, byl vyvinut simulátor (obr. 7) pro testování Surface aplikací na PC. Tento simulátor je už součástí tohoto SDK. Prostředí simulátoru tvoří dvě hlavní části: simulace plochy obrazovky a panel pro ovládání kontaktů.

Plocha obrazovky je místo, které simuluje fyzickou obrazovku MS Surface. Zde probíhá veškerá uživatelská interakce, tzn. pokládání různých druhů kontaktů.

Část, která je zde navíc, je panel pro správu kontaktů. Jedná se o programovou náhradu skutečných objektů. Uživatel si může vybrat jeden ze tří typů vstupu:

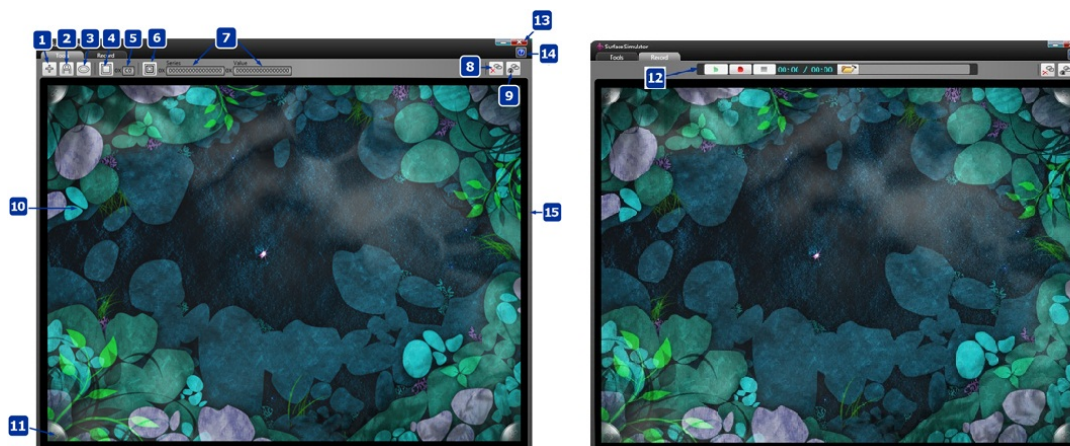
- prst
- skvrna
- označený objekt

Skvrna reprezentuje kontakt neoznačeného objektu. Název vzniknul díky tomu, že snímacímu systému se zobrazí styk objektu jako skvrna (možno vidět na obr. 3). Označené objekty jsou takové, které jsou vybaveny speciální značkou – tagem.

Pokládání kontaktů se provádí pomocí myši počítače. Kurzorem se najede na místo požadovaného umístění. Levým tlačítkem myši se kontakt pokládá. Pravým tlačítkem myši se kontakt zafixuje, aby se mohly, levým tlačítkem myši, upravit jeho parametry: orientace, velikost, případně tvar (u skvrny). V tomto módu se kontakt nebere jako položený na obrazovku. Obnovení do normálního režimu se provede kliknutím levého tlačítka myši mimo modifikovaný kontakt.

Simulace více kontaktů se provádí připojením více myší k počítači. Každá myš rovná se jeden kontakt. Počet myší si simulátor rozpozná sám, tudíž uživatel kromě zapojení myši nemusí nic provádět.

Detailní popis Microsoft Surface simulátoru je uveden v tabulce 1.



Obrázek 7: Microsoft Surface simulátor

### 3.3 Vstupy

Pro využití všech možností, které MS Surface nabízí, je nutné znát všechny druhy vstupů a jejich vlastnosti. Blokové schéma na obrázku 8 zobrazuje vztahy mezi vstupy a prvky, které na ně reagují.

Jak už bylo uvedeno existují tři typy *kontaktů*: prst, objekt (skvrna) a označený objekt. Všechny tyto kontakty mají stejné *události*, a to: položení, přemístění a zvednutí. Na tyto události mohou reagovat ovládací prvky aplikací.

*Gesta* jsou uživatelské interakce, které aplikace rozeznají aniž by věděly k jakým prvkům uživatelského rozhraní se vztahují. Gesta jsou v podstatě takové interakce, které nemění souřadnice  $\{X, Y\}$  položeného kontaktu. Microsoft Surface SDK nabízí události pro dva druhy gest: poklepání a podržení.

Oproti tomu *manipulace* mohou být správně vyloženy jen v kontextu s danými ovládacími prvky uživatelského rozhraní. Např.: aplikace může rozeznat dva prsty pohybující se směrem od sebe jako „roztažení“, dvě nezávislé manipulace „přesunu“ nebo jako několik dalších možností. Konkrétní rozeznání závisí na tom, jak se prvky uživatelského rozhraní jeví v pozadí konkrétní interakce. [7]

Microsoft Surface SDK rozlišuje manipulace v podobě ovládacích prvků, které odpovídají následujícím manipulacím:

- **táhnutí** (drag)
- **sledování** (pan) – pouze u SurfaceScrollView, SurfaceListBox

	<b>název</b>	<b>funkce</b>
1.	<b>Výběr kontaktu</b>	umožňuje manipulaci s kontakty: přemístění, změnu velikosti, roztažení a otočení
2.	<b>Prst</b>	přepnutí do módu simulace prstu
3.	<b>Skvrna</b>	přepnutí do módu simulace objektu
4.	<b>Tag</b>	přepnutí do módu simulace tagu, který využívá hexadecimálních hodnot
5.	<b>Hodnota tagu</b>	slouží k přiřazení hexadecimální hodnoty tagu
6.	<b>Identity tag</b>	přepnutí do módu simulace tagu, který využívá hodnot tvořených sérií a hodnotou
7.	<b>Série/Hodnota identity tagu</b>	slouží k přiřazení série a hodnoty identity tagu
8.	<b>Odstranit všechny kontakty</b>	
9.	<b>Srýt/Zobrazit kontakty</b>	
10.	<b>Oblast simulace</b>	
11.	<b>Přístupové body</b>	tyto body slouží k přechodu mezi aplikacemi a launcherem
12.	<b>Panel pro nahrávání</b>	sada tlačítek pro nahrávání posloupnosti kontaktů (vhodné pro testování)
13.	<b>Minimalizace, Exit</b>	
14.	<b>Nápověda</b>	
15.	<b>Okno Microsoft Surface simulátoru</b>	velikost okna je zafixována na 1224x868 pixelů

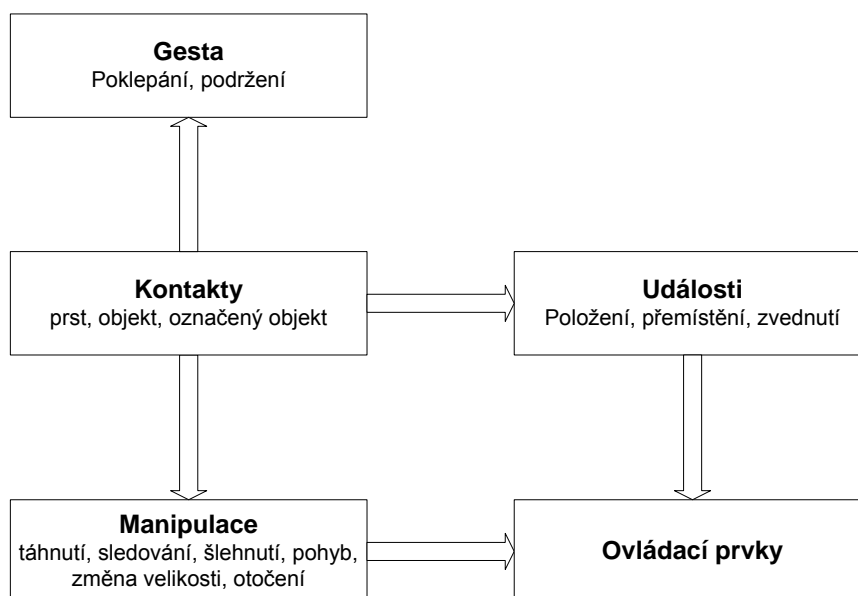
Tabulka 1: Prostředí simulátoru

- **trhnutí** (flick)
- **přesunutí** (move) – pouze u ScatterView
- **změna velikosti** (resize)
- **otočení** (rotate)

### 3.3.1 Označené objekty

Snímací systém Microsoft Surface za pomoci kamer, které pracují v blízkém infračerveném spektru (přesněji 850 nm), pořizuje a zpracovává snímky obrazovky a může tak „vidět“ aktuální obrysy pevných objektů, položených na obrazovku.

Pokud ovšem chcete využívat aplikace řízené objekty, může snímací systém Microsoft Surface také rozpoznat zvláštní označené objekty, které jsou vybaveny speciální stopou



Obrázek 8: Blokové schéma vstupů

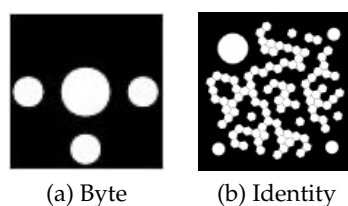
teček nazývanou tag. Tag se skládá z geometrického uspořádání oblastí, které mohou odrážet nebo pohlcovat infračervené světlo.[8]

Označené objekty přidávají aplikacím hned několik výhod nad zpracováním snímků obrazovky.

- Označené objekty jsou výkonnější pro výpočet. Tagy jsou malé a dobře určitelné, takže speciální kód snímacího systému je může lokalizovat a sledovat mnohem rychleji, přesněji a účinněji.
- Platforma Microsoft Surface rozpoznává tagy sama o sobě, takže je jednoduché napsat aplikaci, která je využívá. Pro srovnání aplikace, která pracuje s původními snímky, musí obsahovat svůj vlastní algoritmus na rozpoznávání.
- Každý tag si uchovává jednoznačnou binární hodnotu, aplikace tak může rozeznat jeden objekt od druhého.

Microsoft Surface podporuje dva druhy tagů.

- *Byte tagy* (obr. 9) mají rozsah osmi bitů (1 byte), to umožňuje vytvořit 256 unikátních hodnot. Hodnotu byte tagu v kódu představuje struktura ByteTag.
- *Identity tagy* (obr. 9) mají rozsah 128 bitů (dvě 64-bitové hodnoty), z toho vyplývá relativně nekonečně velký rozsah jedinečných hodnot. Hodnotu identity tagu v kódu reprezentuje struktura IdentityTag.



Obrázek 9: Ukázka tagů

### 3.4 Vývoj aplikací ve vrstvě jádra

*Vrstva jádra* umožňuje přístup k informacím o kontaktech a událostech, proto se na této vrstvě dají vytvářet Microsoft Surface aplikace v jakémkoliv frameworku UI založeného na HWND.

Nejvhodnějším a nejpoužívanějším UI frameworkem je vývojářská platforma Microsoft XNA. XNA umožňuje rychlé vytváření dynamických a sofistikovaných grafických prostředí, podporou komplexního 2D a 3D renderingu.

Dají se ovšem použít i jiné UI frameworky jako Microsoft Managed DirectX nebo Microsoft Windows Forms.[9]

### 3.5 Vývoj aplikací v prezentační vrstvě

*Prezentační vrstva* využívá Microsoft Windows Presentation Foundation (WPF). Jedná se o standartní volbu pro vývoj dotekových aplikací, protože obsahuje vlastní ovládací prvky, defaultně dotekové ovládání a mnoho vzorových prvků UI, jako jsou tlačítka, popisky nebo scroll bary (viz. tabulka 2). WPF také umožňuje použít vlastní grafiku k vytváření aplikací. Programováním aplikací v prezentační vrstvě se můžete zaměřit více na vlastnosti aplikace, zatímco se WPF a předvytvořené ovládací prvky postarají o běžné úkoly.

Prezentační vrstva obsahuje sadu ovládacích prvků WPF ve verzích speciálně pro Microsoft Surface a zároveň podporuje užití XAML souborů pro rychlý vývoj uživatelského rozhraní. Ke zrychlení vývoje, nabízí prezentační vrstva řízené události a připojené vlastnosti, a to způsobem konzistentním s WPF modelem, takže uživatelé kteří už mají zkušenosti s WPF jsou ve výhodě a nemusí se nic nového učit. Prezentační vrstva také nabízí multi-snímací systém, který umožňuje, aby mohl být každý prvek ovládán jedním nebo více kontakty.

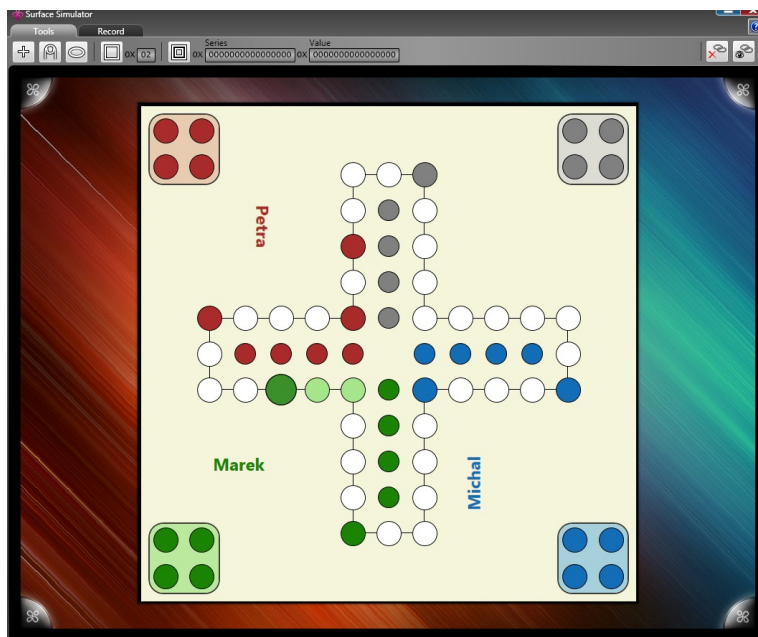
Použitím prezentační vrstvy můžete vytvářet uživatelská rozhraní aplikací pomocí XAMLu a speciálního softwaru na design Microsoft Expression Blend 2 (a vyšší). Pokud skombinujete UI aplikace s korespondujícím C# kódem, můžete tak rychle a snadno vyvíjet dotekové aplikace.[10]

Použity přímo v aplikaci	Komponenty jiných prvků	Základní třídy pro vlastní prvky	Prvky specifické pro Surface
SurfaceWindow SurfaceButton SurfaceInkCanvas SurfaceSlider SurfaceScrollView SurfaceListBox SurfaceTextBox SurfacePasswordBox SurfaceMenu SurfaceContextMenu SurfaceCheckBox SurfaceRadioButton	SurfaceScrollBar SurfaceTrack SurfacePopup SurfaceThumb SurfaceRepeatButton SurfaceToggleButton SurfaceMenuItem SurfaceListBoxItem	SurfaceControl SurfaceContentControl SurfaceUserControl SurfaceItemsControl	ElementMenu LibraryBar LibraryContainer LibraryStack ScatterView ScatterViewItem TagVisualizer TagVisualization

Tabulka 2: Ovládací prvky prezentační vrstvy

## 4 Implementace ukázkové aplikace

Jako ukázkový příklad jsem si vybral implementaci hry „Člověče nezlob se!“ (obr. 10), a to z toho důvodu, že se na této hře dají demonstrovat všechny vlastnosti platformy Microsoft Surface.



Obrázek 10: Aplikace Člověče nezlob se!

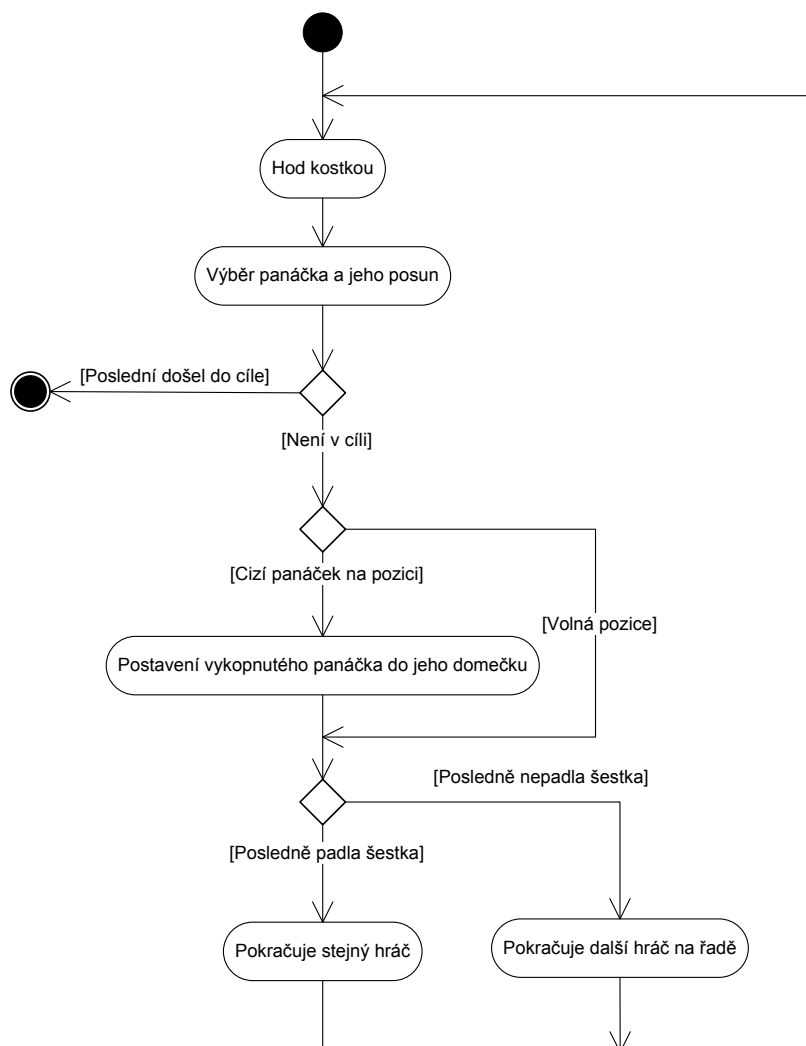
### 4.1 Analýza hry

Pravidla jsou stejná jako u deskové verze této hry, tzn. hra je určena pro 2 až 4 hráče. Každý hráč má 4 figurky. Na začátku se postaví všechny figurky do domečků. Losováním se určí začínající hráč. Pokud má hráč všechny figurky v domečku, pak má právo házet kostkou třikrát, pokud nepadne šestka. Po padnutí šestky si může postavit figurku na start. Od chvíle, kdy má hráč ve hře alespoň jednu figurku, lze uplatnit aktivitní diagram na obrázku 11.

Tzn. hráč hodí kostkou, posune figurku o hozený počet políček, pokud je na políčku figurka jiného hráče, přesune mu ji do jeho domečku. Jestliže hráči posledně padla šestka, tak pokračuje, jinak je na řadě další hráč. Toto se opakuje dokud některý z hráčů nepřesune všechny své figurky do cílových políček. Tím hra končí.

### 4.2 Popis aplikace

K používání této aplikace je potřeba mít šestistennou hrací kostku a figurky pro Člověče nezlob se. Každá figurka musí mít svůj byte tag, a to podle své barvy (viz. tabulka 3).



Obrázek 11: Diagram hry

U figurek se rozlišuje jen barva (na pořadí figurek nezáleží), tudíž figurky stejné barvy mají i stejnou hodnotu tagu.

Na úvodní obrazovce aplikace (obr. 12) se určuje kolik hráčů bude hrát a jaké jsou jejich jména nebo přezdívky. S menu se dá otáčet. Musí být navoleni alespoň dva hráči jinak aplikace nebude pokračovat.

Po zadání počtu hráčů a jejich jmen se zmáčkne tlačítko start a objeví samotná hrací deska (obr. 10). Tu si hráči mohou libovolně natočit a toto natočení posléze potvrdit. Nyní mohou hráči začít pokládat své figurky do domečků. Každá figurka má tag své barvy. Hra sama rozpozná, kdy je umístěna poslední figurka. Pomocí generátoru náhodných čísel se určí začínající hráč. Ten se pozná podle toho, že se mu objeví panel hrací kostky.

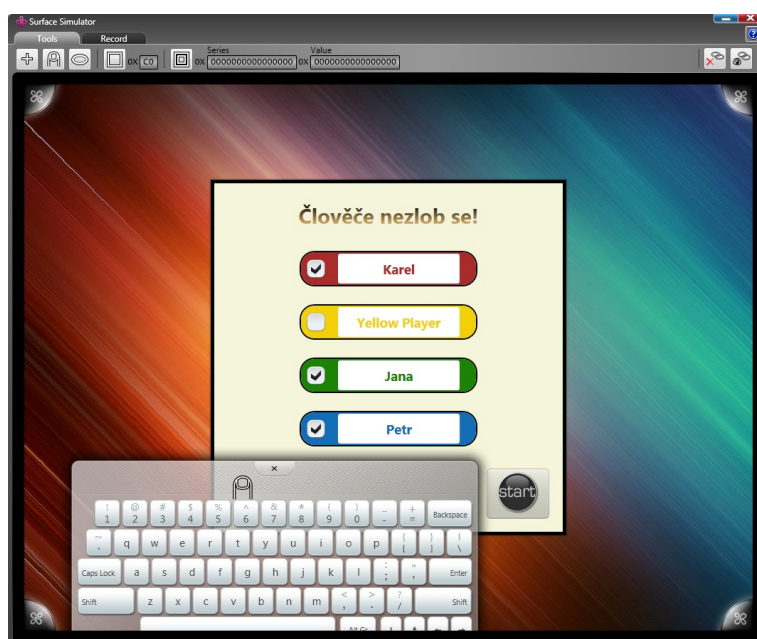


barva figurky	hodnota
žlutá	0x01
modrá	0x02
zelená	0x03
červená	0x04

Tabulka 3: Použité tagy

Hráči hází obyčejnou hrací kostkou a podle výsledku svého hodu zadají hodnotu na panelu herní kostky (prstem). Program na toto zareaguje schováním panelu kostky (v případě hodu šestky panel jen uzamkne), posléze pomocí animací zobrazí možné tahy a bude čekat na vstup od uživatele. Ten musí vybrat jednu svou figurku a tou táhne. Program tah zaznamená a zpřístupní panel kostky dalšímu hráči (v případě šestky odemkne panel aktuálního hráče).

Hra končí oznámením hráče, který jako první přesunul všechny své figurky do cílových políček.



Obrázek 12: Úvodní obrazovka

### 4.3 Analýza aplikace

Aplikace byla vyvinuta v prezentační vrstvě. Tudiž využívá předvytvořené ovládací prvky této vrstvy. GUI této aplikace bylo vytvořeno v Microsoft Expression Blend 3, programově pak bylo upraveno pomocí XAMLu, a to jak v Blend studiu tak i ve Visual

Studiu 2008. Kód, který ošetřuje veškeré programové funkce, byl napsán ve Visual Studiu 2008, kde byla také celá aplikace spojena, tzn. byl zde vytvořen projekt typu Surface Application (WPF).

V příloze A je na obrázku 17 k náhlednutí třídní diagram aplikace, který zobrazuje veškeré vytvořené třídy a jejich propojení.

Po spuštění aplikace se program nejdříve zabývá nastavením parametrů pro hru, jako počet hráčů a jejich jmen, nastavení hrací desky, pořadí hráčů, atd. Po zahájení hry se rozdělí aplikace na dva hlavní procesy: hod kostkou a posun figurky.

Po hodu kostkou (obr. 13) se přičte hozená hodnota k atributům možné pozice všech aktivních figurek hráče (každá figurka má dva hlavní atributy: aktuální pozice a možná pozice). Pokud padla šestka nastaví se všem figurkám v domečku možná pozice na jejich startovní políčko. Poté se zkontroluje zda na možných pozicích už nestojí figurka stejného hráče, v tomto případě by tah nebyl možný, a proto se možná pozice nastaví na stejnou hodnotu jakou má aktuální pozice. Nakonec se schová panel kostky (v případě šestky se jen zablokuje).

Při posunu figurky (obr. 14) se nejdříve ověří zda jde o očekávaný tah (tzn. hráč táhnul svojí figurkou na jedno z možných políček), pokud ne nestane se nic. V opačném případě se potvrdí nová pozice (obsah aktuální pozice se přepíše na obsah možné pozice, u ostatních figurek se obsah možné pozice přepíše na obsah aktuální pozice). Jestliže je políčko obsazeno cizím hráčem nastaví se aktuální pozice figurky cizího hráče na nulu (tzn. je v domečku). Posléze se zruší všechny animace, pokud se jednalo o poslední hráčovu figurku a došla tímto tahem do cílového políčka, pak hra končí oznámením výherce jinak hra pokračuje obnovením panelu kostky dalšího hráče na řadě, v případě že poslední padla šestka pokračuje stejný hráč (odblokuje se mu panel hrací kostky).

## 4.4 Využití SDK

Při vytváření této aplikace byl kladen důraz na co největší využití SDK, aby se tak předvedl veškerý potenciál této platformy. Ovládání aplikace je tedy nejen prsty, ale i označenými objekty.

Tato ukázková aplikace je vyvinuta v prezentační vrstvě, což jí umožňuje přístup už k předvytvořeným ovládacím prvkům, jako např. Surface Button, Surface CheckBox, atd. Na úvodní obrazovce je také využita už předpřipravená klávesnice, která se objeví po najetí prsty na Surface TextBox, což ukazuje, že Microsoft Surface je schopen přijímat i textové vstupy.

Jak menu na úvodní obrazovce, tak hrací deska jsou typu ScatterView Item, což znamená, že se dají libovolně otáčet a přemísťovat. Pro rozpoznávání označených objektů je hrací deska vnořena do prostředí TagVisualizeru.

## 4.5 Možnosti vylepšení aplikace

Tato aplikace není úplně dokonalá. Při návrhu se musel řešit problém jakým způsobem řídit aplikaci. Nejlepším řešením by bylo po hodu kostky, pomocí vlákna s určitým časovým intervalem, kontrolovat pozici všech figurek. Jestli jsou na svých místech nebo jestli

už byl proveden tah. Po provedení tahu by se toto vlákno uspalo a klasicky by se čekalo na vstup hodu kostky, tak jak je to i v této verzi aplikace.

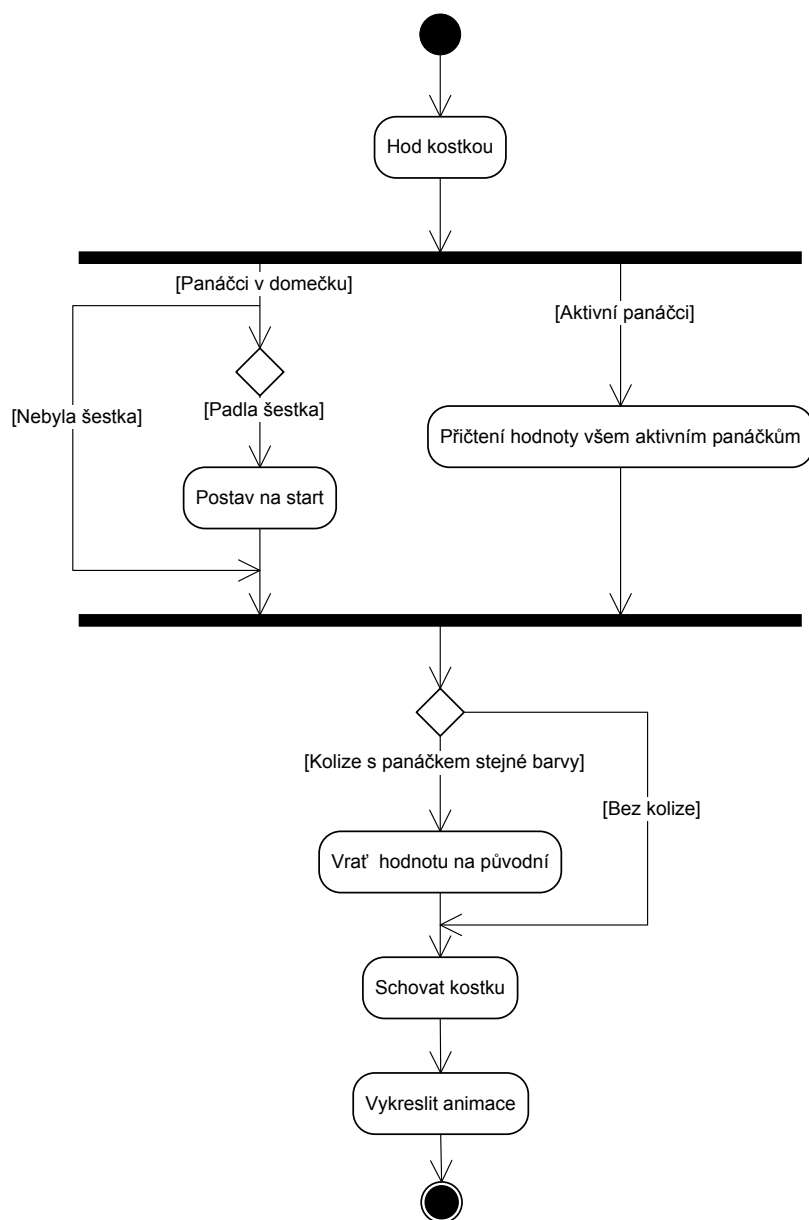
Důvod proč aplikace nebyla takto napsána je ten, že MS Surface SDK neumožňuje aplikaci nastavit daný počet určitých tagů, takže by věděla jaké má čekat vstupy. Namísto toho jediný způsob, jak zjistit polohu nějakého tagu, je pomocí akce tohoto tagu ve spojení s ovládacím prvkem prostředí, např. položení tagu na elipsu. Tzn. že přístup k tagům je proveditelný pouze skrz ovládací prvky.

To by znamenalo, že bysme museli v daném časovém intervalu vlákna zkontrolovat všechny možné interakce všech ovládacích prvků. Oproti tomu v původním případě bysme kontrolovali jen 16 tagů. Výpočetní složitost by tak neúměrně vzrostla.

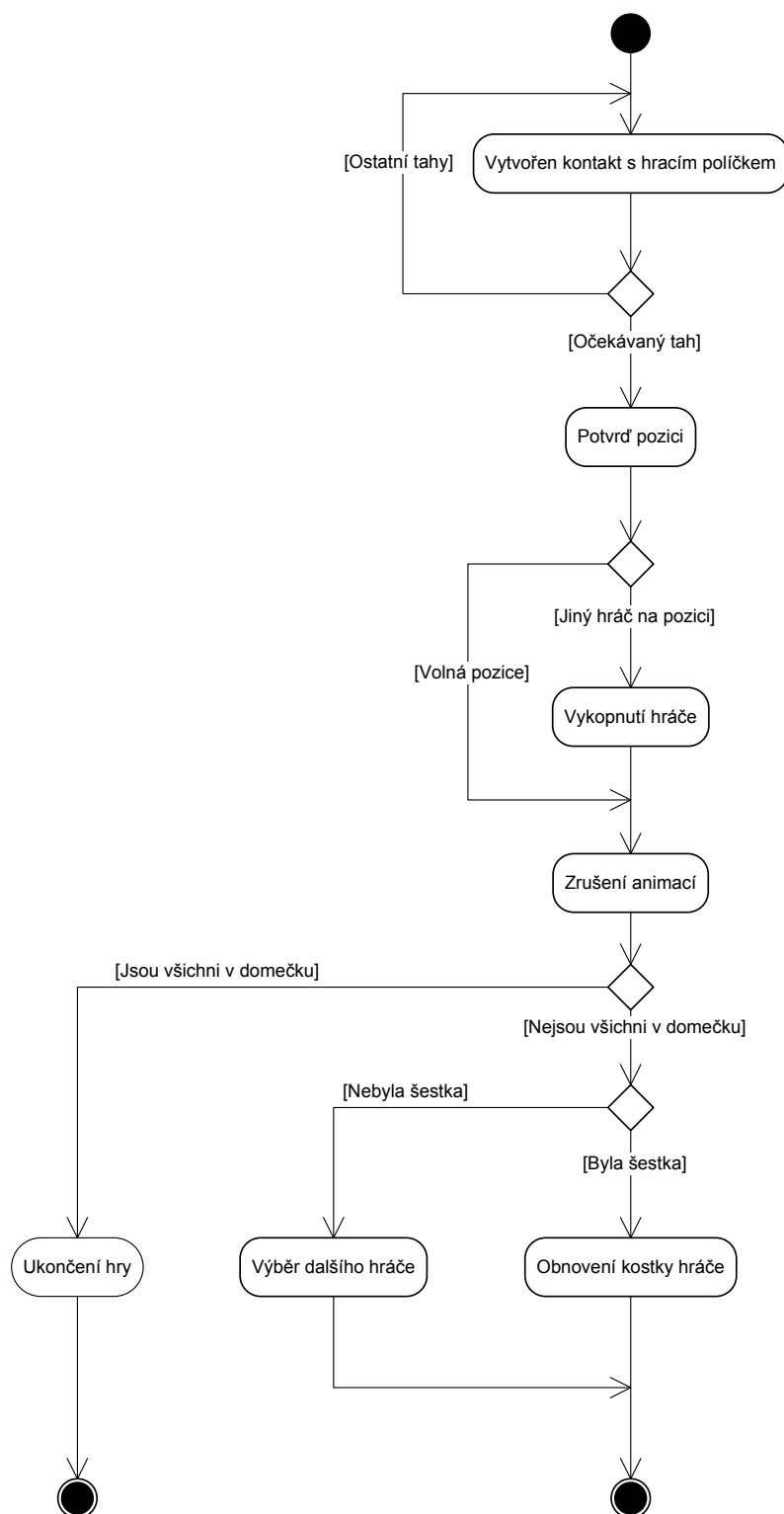
Tato situace by mohla být takový návrh pro vývojeře MS Surface, kteří by mohli vytvořit metodu na kontrolu položených kontaktů a jejich vlastností jako pozice, hodnota, atd.

Další věcí, která by se dala je celkové grafické zpracování. Rozhodně se dají naprogramovat sofistikovanější a bohatější animace. Na samotném vzhledu desky už není moc co vylepšovat, leda vytvořit i hrací desku pro 6 hráčů.

Systém hrací kostky by se dal převézt na reálnou hrací kostku. Ta by měla na každé stěně nalepený tag. A hra by sama po hození načetla hodnotu a zobrazila dané animace. V této verzi to takto provedeno není z technických důvodů. Hra byla vyvíjena i testována na běžném PC, kde by se taková kostka jen velice těžko simulovala.



Obrázek 13: Diagram akcí hodu kostky



Obrázek 14: Diagram akcí posunu figurky

## 5 Specifika vývoje pro Microsoft Surface

MS Surface nabízí nový způsob vztahu, mezi člověkem a technologií, který se intuitivně rozvíjí skrze přirozený lidský kontakt. Použitím designu je tomuto vztahu dán prostor a forma.

Pro navržení přesvědčivých zážitků, musíte o uživatelském rozhraní uvažovat jinak, než při vývoji tradičních GUI. Zejména pokud jste zvyklí pracovat s GUI aplikacemi, založených na ovládání pomocí myši. Existuje mnoho problémů, které musíte vzít v úvahu při vytváření aplikací pro multidotekové systémy, zvláště pak pro MS Surface. Například proto, že MS Surface je platforma pro více uživatelů, nemůžete jednoduše uživatelům dovolit kliknout na nástroj spojený se všemi kontakty na MS Surface jednotce, protože akce jednoho uživatele mohou ovlivnit akce dalších uživatelů.[13]

### 5.1 Tvorba GUI

Jak už bylo uvedeno výše, tvorba GUI pro MS Surface je zcela odlišná od tvorby klasických uživatelských rozhraní pro desktopové aplikace. Vzhledem k tomu, že Surface má tvar stolku, může být uživatelů více a mohou být rozmístěni různě kolem něj a provádět několik věcí najednou. Toto všechno a více by měly navržené aplikace brát v úvahu.

Společnost Microsoft proto sestavila seznam vlastností, které by bylo vhodné zahrnout do návrhu aplikace. Jedná se tedy o následujících osm vlastností.

#### 5.1.1 Přesvědčivý

Přesvědčivé zážitky se dají vytvářet pomocí odstranění nedůvěry uživatelů. Odstranění nedůvěry znamená ochotu lidí přijmout něco jako pravdivé nebo dostatečně skutečné, i přesto že se jedná o věc v reálném světě nemožnou. Kombinace odstranění nedůvěry s interaktivitou, dělá videohry ještě přitažlivější a přibližuje je více k reálnému životu. Z toho důvodu se herní průmysl po mnoho let zaměřuje na vytváření pohlcujících světů simulujících žijící a „dýchající“ prostředí.

A proto i vaše navrhované aplikace pro MS Surface by měly být co nejvíce přesvědčivé a pohlcující.

#### 5.1.2 Společenský

Ve standardních GUI se objevují společenské bariéry, a to z důvodů vstupních a výstupních metod. Při práci s běžnými počítači mají uživatelé k dispozici pouze jednu myš, klávesnici a nedotekový monitor, proto jsou činnosti na těchto počítačích čistě jedno-uživatelské. Oproti tomu Microsoft Surface přijímá více-uživatelské vstupy, takže se kolem něj může sejit více uživatelů a mohou například hrát stejnou instanci hry najednou. Surface tak může povýšit osamělé činnosti na společenské.

Společenské zážitky nejsou omezeny jen na interakce mezi člověkem a rozhraním Surface. Čím méně komunikace mezi jednotlivcem a jednotkou MS Surface, tím více

komunikace probíhá mezi lidmi okolo něj. Lidé se víc zaměří na osoby kolem něj než na počítač, tím pádem se počítač jeví až druhořadý.

V herním průmyslu nahradily soutěživé a spolupracující hry ty tradiční jedno- uživatelské. Konzolové hry zvýšily svou roli prostředníka pro společenskou součinnost, která nastává mezi lidmi spojených hrou. Převzetím kooperačních technik z videoher do MS Surface, můžete aplikace učinit více pohlcující, zábavné a společenské.

### 5.1.3 Prostorový

Tradiční modely interaktivních GUI jsou ploché, tzn. odehrávají se ve 2D. Existují však některé techniky pro vytvoření 2.5-D prostoru (jako úhlové natočení rovin, stínování nebo překrývání prvků), ty mají za efekt vytvoření hloubky v prostředí. Oproti tomu modely interakce MS Surface používají chování reálného světa (3D), takže uživatelé mohou pracovat prostorově ve všech dimenzích.

To ovšem neznamená, že je vždy nutné použít 3D prostředí. Někdy je 3D prostředí kontraproduktivní: dezorientuje a je příliš složité, i přesto by chování a ovládání vašich aplikací mělo vždy brát v potaz existenci osy z. Například fotografie a videa v aplikaci Photos (předinstalována v každém Surface) leží na plochem plátně, ale ve chvíli kdy se jich uživatel dotkne prstem jsou „vytáhnuty“ do popředí, což dodává pocit hloubky a reality.

Microsoft Surface zobrazuje objekty objemově a ovlivňuje tím uživatelské vnímání a prostorovou paměť. Prostředí se dá snadno rozšířit i mimo obrazovku. Uživatelé mohou prostředí libovolně přesouvat, aby tak docílili přemístění obsahu. Objekty mohou být ve 3D prostoru sloučeny a využít tak hloubku k třídění, distribuci nebo zaměření obsahu.

### 5.1.4 Super reálný

Protože dotek je jednoznačně reálný, vytváří pocit přímého ovládání a kontroly nad technologií. To umožňuje vytvářet plynulejší a přirozenější zážitky napodobením skutečných interakcí reálného světa a posunout je tak za hranici možností reality. To sice dělá super realitu přirozenou a přímou, ale ne úplně skutečnou.

Microsoft Surface například umožňuje uživateli dotknout se fotografie a prstem ji posunout, tak jako ve skutečném světě (obr. 15).



Obrázek 15: Posunutí fotografie

Avšak oproti reálnému světu umožňuje MS Surface uživatelům zvětšit či zmenšit fotografii, pomocí dvou prstů (obr. 16).



Obrázek 16: Zvětšení fotografie

Objekty v MS Surface zkušenostech mají „lepší než skutečnou“ kvalitu, která odráží fyzickou podstatu reálných objektů. Tak jako ve skutečném světě, i zde má jakákoliv přímá manipulace okamžitou odezvu. Pokaždé, když prst, ruka nebo objekt působí na MS Surface zkušenost, objeví se odezva, která učiní zkušenost „živou“, přirozenou. Pro navržení takové zkušenosti byste měli objektům umožnit provádět nejen to co by se stalo v reálném světě, ale i něco navíc. Super reálné objekty se super realistickými schopnostmi dávají lidem pocit větší kontroly, než obvyklé způsoby interakce.

Abyste vytvořili přirozené reakce, založte je na interakcích reálného světa a intuitivním způsobem je rozšiřte.

### 5.1.5 Související s prostředím

Zážitky MS Surface odpovídají na doteky lidí a objektů. Průběh těchto reakcí závisí jak na vnitřních, tak i na vnějších vlivech. Ty se mění v závislosti na tom co lidé dělají, kde se ve zkušenostech nacházejí a na jejich chování. (V herním průmyslu se tento vztah mezi příčinou a následkem nazývá kauzalita.)

Zážitky MS Surface by měly reagovat na činnosti uživatelů způsobem, který jasně odpovídá jejich kontextu a chování. Uživatelé tak uvidí, že to co se stalo je přímý následek jejich činnosti, což u nich po nějaké době vyvine smysl pro odhad důsledků jejich chování a naučí se lépe ovládat své jednání. Bez tohoto typu reakcí by se zážitky jevily náhodné a uživatelé by nevěděli jak provést změny v prostředí.

### 5.1.6 Konstruktivní

MS Surface zážitek by neměl nabízet nepřehledné množství možností. Nejlepší způsob je právě jedno prosté řešení, které je zajímavější, zábavnější a hodnotnější. Například tisknutí by mohlo probíhat doslova házením fotografií na vizuální reprezentaci tiskárny.

Použijte ve vašich MS Surface zážitcích konstruktivní myšlení, tím umožníte uživatelům objevovat řešení různých situací zábavným a tvořivým způsobem. Konstruktivní řešení je metoda, která přináší větší výzvu. Nutí uživatele zamýšlet se nad aplikací a ptát se na konstruktivní otázky (např. „Jak tento systém funguje?“), vybízí k řešení menších



problémů (např. „Co se stane, když udělám tohle?“) skrze navádění a poskytování nápo-  
vůd. Konstruktivní řešení nabízí podpůrné vazby, které upřednostňují aktivní průzkum  
místo memorování a opakování. Konstruktivní myšlení můžete použít k navržení MS  
Surface zážitků pomocí optimalizace menších úkolů do zřejmých kroků, tím dojde ke  
zvětšení vizuálních a emočních zážitků.

Další součástí konstruktivního myšlení je nabízení uživatelům jen pár rozumných  
možností najednou, protože uživatelé se cítí zmatení a ochromení, když mají k dispozici  
velký výběr možností. S touto menší nabídkou se zjednoduší rozhodování, odkrývání  
informací nebo počet nutných rozhodnutí za určitou dobu. Tím dojde ke zjednodušení  
ovládání a k většímu procítění práce.

Avšak jednoduchost neznamená zjednodušení. Jednoduché procesy mohou být ne-  
uvěřitelně obsáhlé a mohou mít velký vliv. Zážitky MS Surface by měly demonstrovat  
surové výsledky, které nevyžadují složité operace.

### 5.1.7 Esteticky výkonný

Videohry čím dál více motivují uživatele k prostému průzkumu krásy herního světa. MS  
Surface zážitky by také měly mít vizuální krásu, ale musí také myslet na výkon. Vzhledem  
k fyzické podstatě doteku, gest a přímé manipulace, musí MS Surface aplikace myslet  
na přesvědčivost mezi dotekem objektu a vyvolanou reakcí na obrazovce. V aplikaci  
nesmí dojít ke zpoždění reakce nebo dokonce k zatuhnutí. MS Surface zážitek by měl být  
bezchybný, reagující a okamžitý, aby pocit z něj byl přesný, hladký a přirozený.

K vytvoření vizuálně přitažlivých a výkonných zážitků musí aplikace působit na  
smysly uživatelů.

- **pohled a zvuk** – vytvořte kvalitní vizuální reprezentace obsahu, pohybu a zvuku.
- **představivost** – dovolte uživatelům objevovat nová místa, prvky a efekty, a pod-  
pořte je vizuálně přesvědčivými podněty.
- **výzva** – začleňte výzvy pro uživatele, ale ne skličující problémy, které mohou řešit.
- **přechody** – každý uživatel by měl mít vlastní sadu pravidel, kontextu a cílů, které  
mu umožní najít nejefektivnější cestu k dosažení cíle.
- **vnoření** – navrhnete takové zážitky, které uživatele pohltní. Dovolte jim prozkoumat  
systém, pravidla, kulturu a prostor.

### 5.1.8 Přímý

Zážitky MS Surface reagují na dotek a manipulace jedním nebo více prsty/objekty. Na-  
příklad uživatel může procházet hudební knihovnu otáčením obalů alb, nebo hrát šachy  
se skutečnými figurkami. Činnosti se stávají přechodnými a zpracovanými v závislosti  
na způsobu manipulace obsahu.[13], [14]

## 5.2 Srovnání prezentační vrstvy a vrstvy jádra

Na začátku psaní aplikace, stojí před programátory zásadní rozhodnutí. A to výběr vrstvy, ve které budou aplikaci vyvíjet. Každá vrstva je vhodná pro vývoj jiného druhu aplikací a ovlivňuje možnosti výsledné aplikace.

Prezentační vrstva je vhodná pro psaní MS Surface aplikací, které nevyžadují vysokou úroveň grafiky, jako například komplexní 3D animace. Tato vrstva se spíše hodí, pokud potřebujete rychle vyvinout sofistikovanou aplikaci. Umožňuje totiž využít už předvytvořené funkce a prvky, jako například tlačítka, popisky, vizualizace, atd.

Oproti tomu vrstva jádra umožňuje vytvořit si vlastní programové postupy nebo použití různých frameworků podporujících vývoj 3D aplikací. Na druhou stranu si ale programátor musí napsat vlastní postupy na zpracování kontaktů, událostí apod.

Rozhodnutí v jaké vrstvě pracovat, by měl programátor učinit ještě před zahájením práce. A to důkladným rozбором návrhu vyvíjené aplikace. Ten musí brát v úvahu účel aplikace, její ovládání a cyklus „života“ této aplikace.

## 5.3 Ošetření vstupů

Další specifickou vlastností MS Surface jsou jeho vstupy. Ty jsou zcela odlišné od běžného počítače. Pravděpodobně toto bude důvod proč vyvíjíte aplikaci na MS Surface místo na osobním PC.

Ovšem stejně jako u PC, kde se klávesnice s myší doplňují v práci, tak i vaše MS Surface aplikace by měla skloubit dohromady všechny způsoby ovládání.

## 6 Budoucnost Microsoft Surface

Microsoft Surface je poměrně mladou technologií (první byl uveden do provozu v roce 2008), a proto si ještě musíme počkat na jeho úplnou integraci ve světě informačních technologií. Nicméně společnost Microsoft už oznámila zahájení prací na druhé generaci (přidán projektor na zobrazování obrázků na vrstvě nad obrazovkou, přidány také další senzory na detekování gest a pohybů, bez nutnosti doteku).

Já osobně si myslím, že Microsoft Surface je produkt, který na trhu dosud chyběl, a že je v něm skrytý velký potenciál. Jako příklad možného využití tohoto potenciálu, si dokážu představit třeba podnikatelský záměr „sociálních restaurací“, kde by se jak podávala jídla a nápoje, tak by zde byly přítomny právě stolky Microsoft Surface, ve kterých by byly implementovány různé společenské hry jako např. Osadníci z Katanu, Dostihy a Sázky, Doba Kamenná, hry typu AD&D (už existují verze implementované právě na MS Surface), atd. Ne každý je ochoten si koupit deskovou hru v hodnotě 500-1000 Kč. Navíc ještě potřebujete určitý počet lidí, abyste si mohli hru zahrát. Oproti tomu, když se jde člověk bavit někam do restaurace, nikdy nejde sám, ale v doprovodu několika kamarádů, což už zajišťuje hratelnost.

Další možností je využít silnou stránku MS Surface, kterou je intuitivní ovládání. Stačí se obrazovky dotýkat nebo hýbat s různými předměty. Z toho vyplývá podstatně rychlejší pochopení práce s tímto zařízením. Toho by se dalo využít při seznamování starší generace s výpočetní technikou. Když se tito lidé učí pracovat na běžných počítačích je to ze začátku pro ně nepochopitelné a frustrující. To má za následek nechut' těchto lidí s počítači pracovat. Kdežto kdyby se nejdříve seznámili s touto platformou, naučili by se hravou formou ovládat technologii příbuznou počítači, a snad by tato zkušenost zvětšila jejich chut' do ovládnutí běžného pc.

Momentálně se můžete setkat s Microsoft Surface v několika pobočkách různých známých Amerických řetězců. Bohužel prozatím se dá tato platforma koupit jen na Americkém kontinentě. Ovšem určitě se v blízké době půjde dát koupit i na Evropském trhu.

Shrnutí: Microsoft Surface má slibnou budoucnost a bude zajímavé sledovat, kam až se vyvine.

## 7 Závěr

Mým cílem v bakalářské práci bylo napsání manuálu pro začátečníky s platformou Microsoft Surface.

Osnovu této práce jsem zvolil tak, aby se čtenář postupně dostal od známého k neznámému, tzn. před čtením nemá čtenář žádné zkušenosti s Microsoft Surface, nemusí ani vědět jak vypadá, to se dozví hned na začátku, pokračovat bude ve smyslu běžného uživatele, ze kterého se vyvine v programátora začátečníka.

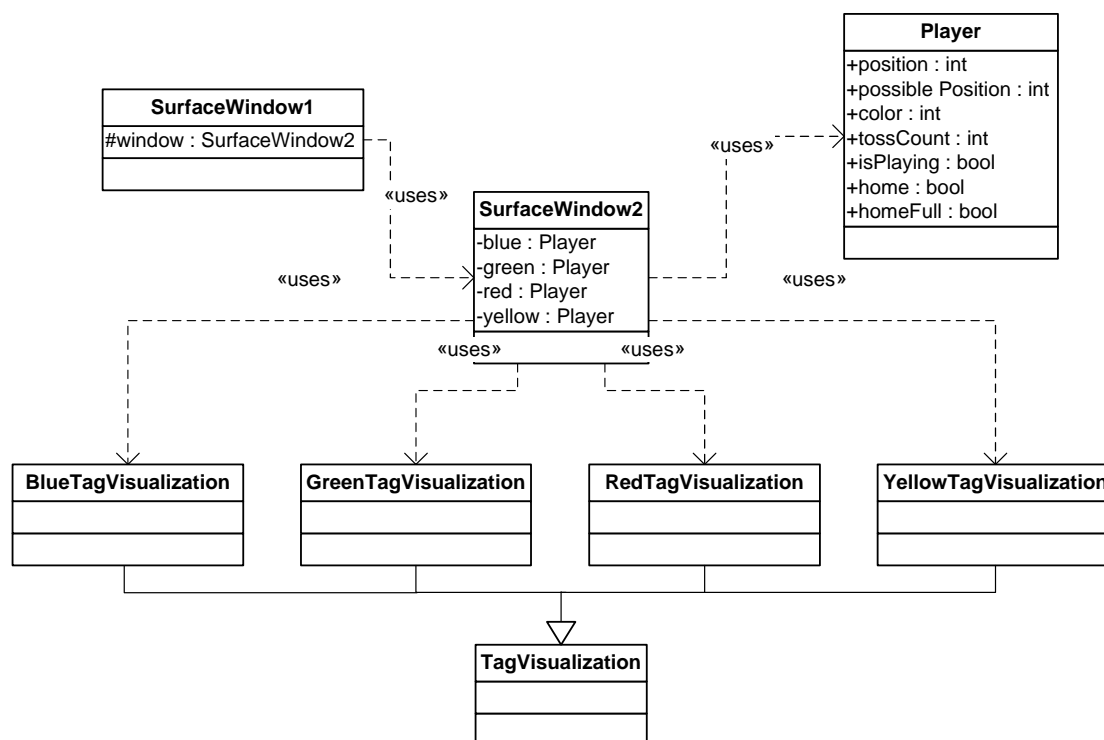
Při psaní této práce jsem přečetl nespočetně článků, ze kterých jsem sestavil celkovou analýzu této platformy. Pro úplné představení možností této platformy jsem napsal ukázkovou aplikaci, jejíž popis je v práci zahrnut.

Téma této bakalářské práce pro mne bylo velice zajímavé, dokonce pro mne její psaní bylo zábavou. V neposlední řadě pro mne měla veliký přínos, a to ten že jsem porozuměl nové technologii a naučil se pro ni vyvíjet aplikace.

## 8 Reference

- [1] *Microsoft Developer Network* [online]. 2010 [cit. 2010-03-21]. Microsoft Surface Architecture. Dostupné z WWW: <<http://msdn.microsoft.com/en-us/library/ee804755.aspx>>
- [2] *Fahad* [online]. 2007 [cit. 2010-04-06]. Microsoft Surface: New Computing Technology In A Table Without Mouse Or Keyboard . Dostupné z WWW: <[http://www.fahad.com/2007\\_07\\_01\\_fahadinc\\_archive.html](http://www.fahad.com/2007_07_01_fahadinc_archive.html)>
- [3] HASSMAN, Martin. *Zdroják* [online]. 23.1.2009 [cit. 2010-03-23]. Microsoft Surface: udělá na vás dojem. Dostupné z WWW: <<http://zdrojak.root.cz/clanky/microsoft-surface-udela-na-vas-dojem>>
- [4] *Microsoft Developer Network* [online]. 2010 [cit. 2010-04-08]. RawImage Visualizer. Dostupné z WWW: <<http://msdn.microsoft.com/en-us/library/ee804802.aspx>>
- [5] DERENE, Glenn. *Popular Mechanics* [online]. 2007 [cit. 2010-03-22]. Microsoft Surface: Behind-the-Scenes First Look (with Video). Dostupné z WWW: <<http://www.popularmechanics.com/technology/industry/4217348.html?page=2>>
- [6] *Microsoft Visual Studio 2008 Documentation* . 2009 [cit. 2010-03-25]. Surface Simulator User Interface.
- [7] *Microsoft Developer Network* [online]. 2010 [cit. 2010-03-26]. Contacts. Dostupné z WWW: <<http://msdn.microsoft.com/en-us/library/ee804917.aspx>>
- [8] *Microsoft Developer Network* [online]. 2010 [cit. 2010-03-26]. Tagged Objects. Dostupné z WWW: <<http://msdn.microsoft.com/en-us/library/ee804823.aspx>>
- [9] *Microsoft Developer Network* [online]. 2010 [cit. 2010-03-26]. Developing Applications with the Core Layer. Dostupné z WWW: <<http://msdn.microsoft.com/en-us/library/ee804882.aspx>>
- [10] *Microsoft Developer Network* [online]. 2010 [cit. 2010-03-27]. Developing Applications with the Presentation Layer. Dostupné z WWW: <<http://msdn.microsoft.com/en-us/library/ee804840.aspx>>
- [11] ARLOW, Jim; NEUSTADT, Illa. *UML 2 a unifikovaný proces vývoje*. 2. dopl. vyd. [s.l.]: Computer Press, 2007. 568 s. ISBN 978-80-251-1503-9.
- [12] PETZOLD , Charles. *Mistrovství ve Windows Presentation Foundation*. Vyd. 1. [s.l.] : Computer Press, 2008. 928 s. ISBN 978-80-251-2141-2.
- [13] *Microsoft Surface Interaction Design Guidelines*. [s.l.] : [s.n.], 2008. 31 s.
- [14] *Microsoft Surface User Experience Guidelines*. [s.l.] : [s.n.], 2009. 77 s.

## A Class Diagram



Obrázek 17: Class Diagram